

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»  
Факультет компьютерных наук  
Образовательная программа «Программная инженерия»**

**СОГЛАСОВАНО**

Научный руководитель, доцент  
факультета компьютерных наук

\_\_\_\_\_ Янович Ю.А.  
«\_\_» \_\_\_\_\_ 2025 г.

**УТВЕРЖДЕНО**

Академический руководитель  
образовательной программы  
«Программная инженерия», старший  
преподаватель департамента  
программной инженерии

\_\_\_\_\_ Н. А. Павлов  
«\_\_» \_\_\_\_\_ 2025 г.

**КОНСТРУКТОР СМАРТ-КОНТРАКТОВ С ДИНАМИЧЕСКОЙ КОМИССИЕЙ  
ДЛЯ ДЕЦЕНТРАЛИЗОВАННЫХ БИРЖ**

**Техническое задание**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.05.01-01 ТЗ 01-1-ЛУ**

Исполнитель:

Студент группы БПИ235

\_\_\_\_\_ / «Фролов И.Г.» /

«\_\_» \_\_\_\_\_ 2025 г.

Инва.№ подп	Подп. и дата	Взам. инв.№	Инва.№ дубл.	Подп. и дата

УТВЕРЖДЕН

RU.17701729.05.01-01 ТЗ 01-1-ЛЮ

**КОНСТРУКТОР СМАРТ-КОНТРАКТОВ С ДИНАМИЧЕСКОЙ КОМИССИЕЙ  
ДЛЯ ДЕЦЕНТРАЛИЗОВАННЫХ БИРЖ**

**Техническое задание**

**RU.17701729.05.01-01 ТЗ 01-1**

**Листов 22**

Инов.№ подп	Подп. и дата	Взам. инв.№	Инов.№ дубл.	Подп. и дата

### АННОТАЦИЯ

Техническое задание – это основной документ, оговаривающий набор требований и порядок создания программного продукта, в соответствии с которым производится разработка программы ее тестирование и приемка.

Настоящее Техническое задание направленное на разработку web-платформы, которая обрабатывает и токенизирует данные о здоровье пользователей содержит следующие разделы: «Введение», «Основания для разработки», «Назначение разработки», «Требования к программе», «Требования к программной документации», «Технико-экономические показатели», «Стадии и этапы разработки», «Порядок контроля и приемки», приложения [7].

В разделе «Введение» указано наименование и краткая характеристика области применения программы.

В разделе «Основания для разработки» указан документ, на основании которого ведется разработка, и наименование темы разработки.

В разделе «Назначение разработки» указано функциональное и эксплуатационное назначение создаваемого программного продукта.

Раздел «Требования к программе» содержит указание на основные требования к функциональным характеристикам программы, к её надежности и к условиям эксплуатации, к составу и параметрам технических средств, к информационной и программной совместимости, к маркировке и упаковке, к транспортировке и хранению, а также специальные требования.

Раздел «Требования к программным документам» содержит указание на предварительный состав программной документации и специальные требования к ней.

Раздел «Технико-экономические показатели» содержит информацию об ориентировочной экономической эффективности разработки, экономические преимущества разработки программы.

Раздел «Стадии и этапы разработки» содержит информацию о стадиях разработки, этапах и содержании работ.

В разделе «Порядок контроля и приемки» указаны общие требования к приемке работы.

Настоящий документ разработан в соответствии с требованиями:

1. ГОСТ 19.101-77 [1]: Виды программ и программных документов.
2. ГОСТ 19.102-77 [2]: Стадии разработки.
3. ГОСТ 19.103-77 [3]: Обозначения программ и программных документов.
4. ГОСТ 19.104-78 [4]: Основные надписи.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.05.01-01 ТЗ 01-1**

5. ГОСТ 19.105-78 [5]: Общие требования к программным документам.
6. ГОСТ 19.106-78 [6]: Требования к программным документам, выполненным печатным способом.
7. ГОСТ 19.201-78 [7]: Техническое задание. Требования к содержанию и оформлению.

Изменения к данному Техническому заданию оформляются согласно ГОСТ 19.603-78 [12], ГОСТ 19.604-78 [13].

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

## СОДЕРЖАНИЕ

1. ВВЕДЕНИЕ .....	6
1.1. Наименование программы .....	6
1.2. Краткая характеристика области применения программы .....	6
2. ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ .....	7
2.1. Документ(ы), на основании которого(ых) ведется разработка .....	7
2.2. Наименование темы разработки .....	7
3. НАЗНАЧЕНИЕ РАЗРАБОТКИ .....	8
3.1. Функциональное назначение .....	8
3.2. Эксплуатационное назначение .....	8
4. ТРЕБОВАНИЯ К ПРОГРАММЕ .....	9
4.1. Требования к функциональным характеристикам .....	9
4.1.1. Требования к составу выполняемых функций .....	9
4.1.2. Требования к организации входных данных .....	10
4.1.3. Требования к организации выходных данных .....	10
4.1.4. Требования к временным характеристикам .....	10
4.1.5. Требования к интерфейсу .....	10
4.2. Требования к надежности .....	10
4.3. Условия эксплуатации .....	10
4.4. Требования к составу и параметрам технических средств .....	10
4.5. Требования к информационной и программной совместимости .....	10
4.5.1. Требования к информационным структурам и методам решения .....	10
4.5.2. Требования к исходным кодам и языкам программирования .....	11
4.5.3. Требования к программным средствам, используемых программой .....	11
4.5.4. Требования к защите информации и программ .....	11
4.6. Требования к маркировке и упаковке .....	11
4.7. Требования к транспортированию и хранению .....	12
4.8. Специальные требования .....	12
5. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ .....	13
5.1. Состав программной документации .....	13
5.2. Специальные требования к программной документации .....	13
6. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ .....	14

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.05.01-01 ТЗ 01-1**

6.1. Предполагаемая потребность .....	14
6.2. Целевая аудитория .....	14
6.3. Экономические преимущества разработки по сравнению с отечественными и зарубежными аналогами .....	14
7. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ .....	16
7.1. Стадии разработки, этапы и содержание работ .....	16
7.2. Сроки разработки и исполнители .....	18
8. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ .....	19
8.1. Виды испытаний .....	19
8.2. Общие требования к приемке работы .....	19
ПРИЛОЖЕНИЕ 1. СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ .....	20
ПРИЛОЖЕНИЕ 2. ССЫЛКИ НА АНАЛОГИ .....	22

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

## 1. ВВЕДЕНИЕ

### 1.1. Наименование программы

Наименование программы — «Конструктор смарт-контрактов с динамической комиссией для децентрализованных бирж».

Наименование программы на английском языке — «Constructor of Dynamic Fees for Decentralized Exchanges».

### 1.2. Краткая характеристика области применения программы

Проект реализует библиотеку смарт контрактов с динамическими комиссиями для DEX на основе Uniswap V4, позволяющий автоматически корректировать размер комиссии в зависимости от рыночных условий. Основная область применения программы - оптимизация управления децентрализованными биржами (DEX) своих пулов ликвидности и предоставление более гибких условий для пользователей и поставщиков ликвидности.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

## 2. ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ

### 2.1. Документ(ы), на основании которого(ых) ведется разработка

Разработка ведётся на основании учебного плана подготовки бакалавров по направлению 09.03.04 «Программная инженерия» и утвержденной академическим руководителем программы темы курсового проекта.

### 2.2. Наименование темы разработки

Наименование темы разработки: «Конструктор смарт-контрактов с динамической комиссией для децентрализованных бирж».

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата



### 3. НАЗНАЧЕНИЕ РАЗРАБОТКИ

#### 3.1. Функциональное назначение

Данный продукт представляет из себя библиотеку смарт-контрактов для DEX на основе фреймворка Uniswap V4 (хуков), а именно реализацию функций::

- 1) Динамический расчет и автоматическая корректировка размера комиссии пула ликвидности в зависимости от заданного алгоритма и рыночных условий.
- 2) Интеграция с оракулом Chainlink для получения актуальных ценовых данных с централизованных бирж (CEX) для алгоритмов комиссий, основанных на внешней информации.
- 3) Взаимодействие с ключевыми артефактами экосистемы Uniswap V4: PoolManager, SwapRouter, PositionManager для создания пулов, управления ликвидностью и выполнения обменных операций (свопов).
- 4) Предоставление унифицированного интерфейса для всех реализованных типов хуков динамических комиссий, обеспечивающего их совместимость и простоту интеграции.
- 5) Сквозное тестирование логики работы смарт-контрактов на локальном блокчейне (Anvil) с разворачиванием всей необходимой инфраструктуры и симуляцией торговой активности.
- 6) Развертывание смарт-контрактов на тестовые и основные сети блокчейна через специализированные скрипты, включающие подготовку всех зависимых артефактов.
- 7) Предоставление интерфейса для конструирования смарт-контракта хука на основе выбранного пользователем алгоритма расчета комиссии.

#### 3.2. Эксплуатационное назначение

Продукт создан для снижения непостоянных потерь (impermanent loss) поставщиков ликвидности и повышения эффективности децентрализованных бирж (DEX). Основными конечными потребителями разрабатываемой библиотеки являются разработчики децентрализованных финансовых (DeFi) протоколов, команды, создающие специализированные DEX, а также поставщики ликвидности, стремящиеся оптимизировать свою доходность.

При использовании локального блокчейна требуется серверное оборудование, соответствующее техническим требованиям, и доступ к сети Интернет.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

#### 4. ТРЕБОВАНИЯ К ПРОГРАММЕ

##### 4.1. Требования к функциональным характеристикам

##### 4.1.1. Требования к составу выполняемых функций

Программа должна иметь реализацию следующего функционала:

А. Реализация библиотеки смарт-контрактов (хуков) динамических комиссий для Uniswap V4

- Реализовать базовое наследование от контракта BaseOverrideFee с унифицированным интерфейсом для всех типов динамических комиссий
- Реализовать несколько конкретных типов хуков, каждый со своей математической моделью расчета комиссии (например, на основе волатильности, объема торгов, арбитражного спреда)
- Для алгоритмов, использующих внешние данные, реализовать интеграцию с оракулом Chainlink для получения цен с централизованных бирж (CEX)
- Реализовать механизм автоматической корректировки размера комиссии пула в зависимости от текущих рыночных условий
- Реализовать совместимость хуков с основными артефактами Uniswap V4: PoolManager, SwapRouter, PositionManager

В. Реализация инфраструктуры для разработки и тестирования

- Реализовать утилиты для развертывания всей необходимой инфраструктуры Uniswap V4 на локальном блокчейне (Anvil)
- Реализовать скрипты для автоматического развертывания хуков и создания ликвидностных пулов с ними
- Реализовать комплексные тесты, симулирующие поведение пользователя и изменение комиссии

С. Реализовать веб-интерфейс для конструирования и развертывания смарт-контрактов

- Интерфейс предоставляет выбор типа динамической комиссии из доступных алгоритмов
- При выборе типа комиссии отображается генерируемый код смарт-контракта и его основные параметры
- Предоставляется возможность скачать готовый проект с выбранным хуком, включая все необходимые зависимости и инструкции по развертыванию
- Для протестированных алгоритмов отображается сравнительный анализ эффективности на исторических данных (если такой анализ проведен)

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

#### 4.1.2. Требования к организации входных данных

Библиотека принимает входные параметры через командную строку и аргументы конструктора смарт-контрактов. Для веб-интерфейса: тип алгоритма расчета комиссии выбирается пользователем из списка на веб-странице.

#### 4.1.3. Требования к организации выходных данных

Основными выходными данными являются сгенерированные смарт-контракты и необходимые зависимости для их развертывания. Для скриптов тестирования в командную строку выводятся детализированные логи транзакций и изменения состояния блокчейна во время тестов.

#### 4.1.4. Требования к временным характеристикам

Требования к временным характеристикам программы не предъявляются.

#### 4.1.5. Требования к интерфейсу

Интерфейс смарт-контрактов реализуется через командную строку и должен быть понятен разработчикам клиентской части платформы.

#### 4.2. Требования к надежности

Программа не должна завершаться аварийно при любых действиях пользователя.

#### 4.3. Условия эксплуатации

Климатические условия эксплуатации должны соответствовать требованиям к техническим средствам, реализующим данный программный продукт.

#### 4.4. Требования к составу и параметрам технических средств

Для использования программы к компьютеру предъявляются следующие требования:

- Объем оперативной памяти не менее 32 МБ
- Компьютер обладает устройствами ввода (клавиатурой, мышью / трекпадом)
- Компьютер обладает устройствами вывода (монитор)
- На компьютере установлена операционная система Windows или MacOS.
- На компьютере установлен браузер Chrome или Arc.
- Установлен блокчейн-фреймворк «Forge».

#### 4.5. Требования к информационной и программной совместимости

##### 4.5.1. Требования к информационным структурам и методам решения

Реализация динамических комиссий основана на хуковой системе Uniswap V4. Расчет комиссий происходит с использованием математических моделей, учитывающих рыночные параметры. Для

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

алгоритмов, зависящих от внешних данных, используется оракул Chainlink. Тестирование эффективности проводится через симуляцию торговли на исторических данных.

#### 4.5.2. Требования к исходным кодам и языкам программирования

##### 1. Смарт-контракты (хуки) и инфраструктура

- Основная разработка на языке Solidity с использованием фреймворка Foundry (Forge)
- Хуки реализуют интерфейсы Uniswap V4 Hooks и наследуют OpenZeppelin библиотеки
- Для локального тестирования и развертывания используется Anvil (локальная EVM из Foundry)

##### 2. Внешние данные и оракулы

- Интеграция с Chainlink Price Feeds для получения ценовых данных с CEX
- Использование Binance Market Data API для сбора исторических данных

##### 3. Тестирование и анализ

- Написание тестов на Solidity с использованием Forge
- Анализ эффективности на исторических данных с использованием Python

##### 4. Веб-интерфейс

- Интерфейс для конфигурации контрактов реализован на Python (Flask или подобный фреймворк)
- Генерация и отображение Solidity кода через веб-интерфейс

#### 4.5.3. Требования к программным средствам, используемым программой

Версии используемых технологий

- Solidity версии 0.8.0 и выше (совместимость с Uniswap V4)
- Toolkit Foundry (Forge, Cast, Anvil) с Forge версии 1.12.0 и выше
- FoundryUp версии 1.4.0 и выше (для управления версиями Foundry)
- Uniswap V4 Core версии 4.0.0 и Periphery contracts
- Chainlink версии 0.2.7 и выше
- OpenZeppelin UniswapHooks библиотека версии 1.1.1 и выше
- Python версии 3.10 и выше

#### 4.5.4. Требования к защите информации и программ

Требования к защите информации и программ не предъявляются.

#### 4.6. Требования к маркировке и упаковке

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

Программа распространяется через сеть Интернет. Программа распространяется в виде электронного пакета, содержащего программную документацию, приложение (исполняемые файлы и прочие необходимые для работы файлы, в том числе файлы с исходным кодом) и презентацию проекта. Специальных требований к маркировке и упаковке не предъявляется.

#### 4.7. Требования к транспортированию и хранению

Транспортировка программного продукта должна осуществляться без нарушения полноты комплекта, предоставленного разработчиком изначально.

Хранение продукта осуществляется с помощью репозитория на GitHub.

#### 4.8. Специальные требования

Специальные требования к программе не предъявляются.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

## 5. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

### 5.1. Состав программной документации

- 1) «Конструктор смарт контрактов с динамической комиссией для децентрализованных бирж». Техническое задание (ГОСТ 19.201-78 [7]).
- 2) «Конструктор смарт контрактов с динамической комиссией для децентрализованных бирж». Пояснительная записка (ГОСТ 19.404-79 [10]).
- 3) «Конструктор смарт контрактов с динамической комиссией для децентрализованных бирж». Программа и методика испытаний (ГОСТ 19.301-79 [8]).
- 4) «Конструктор смарт контрактов с динамической комиссией для децентрализованных бирж». Текст программы (ГОСТ 19.401-78 [9]).
- 5) «Конструктор смарт контрактов с динамической комиссией для децентрализованных бирж». Руководство программиста (ГОСТ 19.504-79 [11]).

### 5.2. Специальные требования к программной документации

1. Документы к программе должны быть выполнены в соответствии с ГОСТ 19.106-78 [6] и ГОСТами к каждому виду документа (см. п. 5.1.);
2. Пояснительная записка должна быть загружена в систему Антиплагиат через LMS «НИУ ВШЭ».
3. Документация и программа сдается в электронном виде в формате .pdf или .docx. в архиве формата .zip или .rar;
4. За две недели до защиты комиссии все материалы курсового проекта:
  - программная документация,
  - программный проект,
  - исполняемый файл,
  - отчет системы Антиплагиат

должны быть загружены одним или несколькими архивами в проект дисциплины «Курсовой проект» в личном кабинете в информационной образовательной среде SmartLMS НИУ ВШЭ.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

## 6. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

### 6.1. Предполагаемая потребность

Библиотека будет использоваться как решение для повышения эффективности децентрализованных бирж (DEX) и защиты поставщиков ликвидности. Внедрение динамических комиссий позволит снизить непостоянные потери (impermanent loss), что повысит общую доходность пулов ликвидности.

Функционал библиотеки будет востребован разработчиками DeFi-протоколов, командами, запускающими специализированные DEX, и профессиональными поставщиками ликвидности. Ориентировочная экономия для крупного пула ликвидности может составлять от 1 000 до 10 000 долларов в месяц за счет снижения непостоянных потерь и оптимизации комиссионных доходов.

### 6.2. Целевая аудитория

Функционал программы будет использоваться разработчиками DeFi-протоколов, командами, запускающими специализированные DEX, и профессиональными поставщиками ликвидности.

### 6.3. Экономические преимущества разработки по сравнению с отечественными и зарубежными аналогами

Для оценки преимуществ проекта было проведено сравнение функциональных характеристик программы с аналогами. Данные сравнения отображены в таблице 1.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

	Pancake Swap	Trader Joe LB	Uniswap V4	Camelot DEX	THENA/Algebra	Разработка
Динамические комиссии	+	+	+	+	+	+
Глубокая кастомизация алгоритмов	-	-	-	-	-	+
Снижение ИЛ как приоритетная цель	-	+	-	-	-	+
Интеграция с Chainlink оракулами	-	-	-	±	+	+
Готовые шаблоны/библиотеки	+	-	-	-	+	+
Анализ на исторических данных	-	-	-	-	-	+
Открытый исходный код	+	+	+	+	+	+
Интеграция с хуками Uniswap V4	-	-	+	-	-	+
Итого	3	3	3	2.5	4	8

Таблица 1. Сравнение функциональных характеристик с аналогами

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата



## 7. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ

## 7.1. Стадии разработки, этапы и содержание работ

Стадии и этапы разработки были выявлены с учётом ГОСТ 19.102-77 [2].

Стадия разработки	Этапы разработки	Содержание работ	Исполнитель	Сроки выполнения
Техническое задание	Обоснование необходимости разработки	Постановка задачи	Фролов И.Г.	06.11.2025
		Сбор теоретических материалов	Фролов И.Г.	10.11.2025 - 20.11.2025
	Разработка технического задания	Определение требований к программе	Фролов И.Г.	20.11.2025 - 30.11.2025
		Определение стадий, этапов и сроков разработки программы и документации на нее	Фролов И.Г.	30.11.2025 - 04.12.2025
		Согласование технического задания с научным руководителем	Фролов И.Г.	04.12.2025 - 16.12.2025
		Загрузка разработанного технического задания в SmartLMS	Фролов И.Г.	16.12.2025

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

## RU.17701729.05.01-01 ТЗ 01-1

Стадия разработки	Этапы разработки	Содержание работ	Исполнитель	Сроки выполнения
Рабочий проект	Разработка программы	Смарт-контракты	Фролов И.Г.	04.12.2025 - 29.01.2026
		Согласование основной части кода с научным руководителем	Фролов И.Г.	29.01.2026 - 04.02.2026
		Сдача основной части кода в SmartLMS	Фролов И.Г.	04.02.2026
	Разработка программной документации	Разработка программных документов в соответствии с требованиями ГОСТ 19.101-77	Фролов И.Г.	04.02.2026 - 31.03.2026
	Испытания программы	Разработка, согласование и утверждение порядка испытаний	Фролов И.Г.	04.02.2026 - 15.02.2026
		Проведение испытаний программы в соответствии с утвержденными методиками	Фролов И.Г.	15.02.2026 - 31.03.2026
		Корректировка программы и документации по результатам испытаний	Фролов И.Г.	15.02.2026 - 31.03.2026

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

Стадия разработки	Этапы разработки	Содержание работ	Исполнитель	Сроки выполнения
Внедрение	Подготовка и передача программы	Представление разработанного продукта научному руководителю и получение отзыва	Фролов И.Г.	05.04.2026
		Подготовка программы и документации для презентации и защиты	Фролов И.Г.	в процессе договоренностей
		Загрузка пояснительной записки в систему Антиплагиат через ЛМС НИУ ВШЭ	Фролов И.Г.	в процессе договоренностей
		Загрузка материалов курсового проекта в ЛМС	Фролов И.Г.	в процессе договоренностей
		Защита программного продукта комиссии	Фролов И.Г.	в процессе договоренностей

## 7.2. Сроки разработки и исполнители

Программный продукт должен завершен не позднее 4 апреля 2026 года.

Исполнитель – Фролов Иван Григорьевич, студент группы БПИ235 Факультета Компьютерных Наук НИУ ВШЭ.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

## 8. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ

### 8.1. Виды испытаний

Проверка продукта на соответствие техническому заданию, а также другим утвержденным требованиям может происходить на любой стадии разработки и может включать в себя один или несколько видов испытаний:

1. Как полное, так и частичное функциональное тестирование
2. Тестирование производительности
3. Тестирование удобства пользования
4. Тестирование безопасности.

### 8.2. Общие требования к приемке работы

Контроль и приемка разработки осуществляются в соответствии с документом «Программа и методика испытаний» (ГОСТ 19.301–79).

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

**ПРИЛОЖЕНИЕ 1. СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ**

1. ГОСТ 19.101-77: Виды программ и программных документов. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
2. ГОСТ 19.102-77: Стадии разработки. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
3. ГОСТ 19.103-77: Обозначения программ и программных документов. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
4. ГОСТ 19.104-78: Основные надписи. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
5. ГОСТ 19.105-78: Общие требования к программным документам. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
6. ГОСТ 19.106-78: Требования к программным документам, выполненным печатным способом. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
7. ГОСТ 19.201-78: Техническое задание. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
8. ГОСТ 19.301-79: Программа и методика испытаний. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
9. ГОСТ 19.401-78: Текст программы. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
10. ГОСТ 19.404-79: Пояснительная записка. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
11. ГОСТ 19.505-79: Руководство оператора. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
12. ГОСТ 19.603-78: Общие правила внесения изменений. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
13. ГОСТ 19.604-78: Правила внесения изменений в программные документы, выполненные печатным способом. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
14. Документация Solidity. Электронный ресурс. URL: <https://docs.soliditylang.org/en/latest/>
15. Документация Foundry (Forge, Anvil, Cast). Электронный ресурс. URL: <https://book.getfoundry.sh/>
16. Документация FoundryUp. Электронный ресурс. URL: <https://github.com/foundry-rs/foundry/tree/master/foundryup>

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

17. Uniswap V4 Hooks Guide. Электронный ресурс. URL: <https://docs.uniswap.org/contracts/v4/overview>
18. Uniswap V4 Periphery. Электронный ресурс. URL: <https://github.com/Uniswap/v4-periphery>
19. Документация Chainlink Data Feeds. Электронный ресурс. URL: <https://docs.chain.link/data-feeds>
19. OpenZeppelin UniswapHooks. Электронный ресурс. URL: <https://github.com/OpenZeppelin/uniswap-hooks>
20. Binance Market Data API. Электронный ресурс. URL: <https://developers.binance.com/docs/binance-spot-api-docs/rest-api/market-data-endpoints#uiklines>
21. Python Documentation. Электронный ресурс. URL: <https://docs.python.org/3/>

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

## ПРИЛОЖЕНИЕ 2. ССЫЛКИ НА АНАЛОГИ

Приложение	Ссылка
PancakeSwap	<a href="https://docs.pancakeswap.finance/">https://docs.pancakeswap.finance/</a>
Trader Joe Liquidity Book	<a href="https://traderjoexyz.com/liquidity-book">https://traderjoexyz.com/liquidity-book</a>
Maverick Protocol	<a href="https://www.mav.xyz/">https://www.mav.xyz/</a>
Uniswap V4	<a href="https://docs.uniswap.org/contracts/v4/overview">https://docs.uniswap.org/contracts/v4/overview</a>
Camelot DEX	<a href="https://docs.camelot.exchange/protocol/amm-v2/dynamic-directional-fees">https://docs.camelot.exchange/protocol/amm-v2/dynamic-directional-fees</a>
THENA	<a href="https://docs.thena.fi/thena/liquidity-provision/concentrated-liquidity">https://docs.thena.fi/thena/liquidity-provision/concentrated-liquidity</a>
QuickSwap	<a href="https://blog.quickswap.exchange/posts/quickswap-just-got-smarter-with-dynamic-fee-hooks-on-soneium">https://blog.quickswap.exchange/posts/quickswap-just-got-smarter-with-dynamic-fee-hooks-on-soneium</a>
Algebra Integral	<a href="https://docs.algebra.finance/algebra-integral-documentation/user-guide-template-for-dexes/concentrated-liquidity-and-modular-architecture-basics/fee-mechanics/dynamic-fee">https://docs.algebra.finance/algebra-integral-documentation/user-guide-template-for-dexes/concentrated-liquidity-and-modular-architecture-basics/fee-mechanics/dynamic-fee</a>

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.01-01 ТЗ 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

## ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

[illegible]